WHAT IS CLAIMED IS

Sub A1

1. A memory access method for a multiprocessor system which includes a plurality of system modules coupled via a crossbar module, each of the system modules including a buffer which holds data and a plurality of processors having a cache memory which temporarily holds data, said memory access method comprising:

a step, responsive to a read request from a processor within an arbitrary system module, holding data preread from a system module other than the arbitrary system module in a buffer within the crossbar module.

2. The memory access method as claimed in claim 1, further comprising:

a step of setting information indicating whether or not to carry out a data preread with respect to the arbitrary system module, depending on a program which is executed by one or a plurality of processors within the arbitrary system module.

3. The memory access method as claimed in claim 2, further comprising:

a step of adding, to a data transfer of the preread data, a priority which is lower than a priority of a normal data transfer.

Cont A1

4. The memory access method as claimed in claim 1, further comprising:

a step of adding, to a data transfer of the preread data, a priority which is lower than a priority of a normal data transfer.

5. A multiprocessor system comprising:

a plurality of system modules;

at least one crossbar module; and

a bus coupling the system modules and the crossbar module,

each of the system modules including a buffer which holds data, a plurality of processors each having a cache memory which temporarily holds data, and a control unit which controls input and output of data with respect to the system module to which the control unit belongs,

a data transfer between two system modules being made via the crossbar module,

said crossbar module including a buffer which holds data preread from a system module other than an arbitrary system module in responsive to a read request from a processor within the arbitrary system module.

6. The multiprocessor system as claimed in claim 5, wherein the arbitrary system module includes means for setting information indicating whether or not to carry out a data preread with respect to the arbitrary system module, depending on a program which is executed by one or a plurality of

Cont A1

processors within the arbitrary system module.

7. The multiprocessor system as claimed in claim 6, wherein each of the system modules further includes means for adding, to a data transfer of the preread data, a priority which is lower than a priority of a normal data transfer.

8. The multiprocessor system as claimed in claim 5, wherein each of the system modules further includes means for adding, to a data transfer of the preread data, a priority which is lower than a priority of a normal data transfer.

9. The multiprocessor system as claimed in claim 5, wherein one of the system modules, which has a memory with a requested address of the read request, includes means for starting a data preread at a timing before detecting a state of the cache memory included therein.

10. The multiprocessor system as claimed in claim 5, wherein:

the plurality of system modules, the crossbar module, and the bus form a node; and

a plurality of nodes are coupled via the crossbar module of adjacent nodes.

Cont A1

11. A multiprocessor system comprising:

a plurality of nodes each including a plurality of system modules, a crossbar module, and a bus coupling the system modules and the crossbar module within each node; and

a bus coupling adjacent nodes via the crossbar modules of the adjacent nodes,

each of the system modules including a buffer which holds data, a plurality of processors each having a cache memory which temporarily holds data, and a control unit which controls input and output of data with respect to the system module to which the control unit belongs,

a data transfer between two system modules being made via at least one crossbar module,

said crossbar module including a buffer which holds data preread from a system module other than an arbitrary system module in responsive to a read request from a processor within the arbitrary system module.

Add A1